Managing an EU-funded Research Project – Case Study on AMALTHEA

Robert Höttger, Daniel Fruhner, Burkhard Igel, Carsten Wolff

robert.hoettger@fh-dortmund.de, daniel.fruhner@fh-dortmund.de, igel@fh-dortmund.de, carsten.wolff@fh-dortmund.de
Research Focus Group PIMES
University of Applied Sciences and Arts
FH Dortmund, Germany
Sonnenstraße 96
44139 Dortmund

Keywords: EU-funded research management, collaborative work, automotive, model driven software development, multicore, toolchain

Abstract: The challenge of managing an EU-funded research project lies in the efficient coordination, agreement and common arrangement of tasks, interfaces or simply work load within partners from different countries, companies or institutions and thus a specific field of expertise. Not only the experience or depth of current research behavior but especially available resources and predefined deadlines must be considered during any process of development or coordination point. Hence management includes the planning and controlling for the purpose of accomplishing a discussed intent and furthermore implies comprehensive knowledge about concrete details according to who is working on what issue for what timeframe with the purpose to achieve what information, data or any specific goal. This paper forms a description of a management system including hierarchical structures and various tools in order to achieve a stabilized organization with well defined structures within the AMALTHEA project.

1. Introduction to AMALTHEA

The goal of the ITEA2 [7] Amalthea research project is a consistent, open and expandable tool platform for automotive embedded-multicore-system engineering based on model-driven methodology. There are some specific features the platform has to support. One feature is the AUTOSAR compatibility, because AUTOSAR is the standardized automotive software architecture manufacturer and suppliers have to respect. Another aspect, that has to be integrated, is the support of multi-core systems. Due to the various functional requirements, most single-core processors are computing at their performance limits and less capable of the variety and complexity of the increasing number of computation intensive systems. Productline-engineering shall extend the Amalthea platform in order to support the advantage of efficient variant-handling within the automotive development environment. In order to provide a common tool development basis, the eclipse environment as well as two eclipse based frameworks (TA Simulator [4], Yakindu SCT / Damos [1]) are chosen. Interfaces shall be developed and benefits of each tool shall be combined.

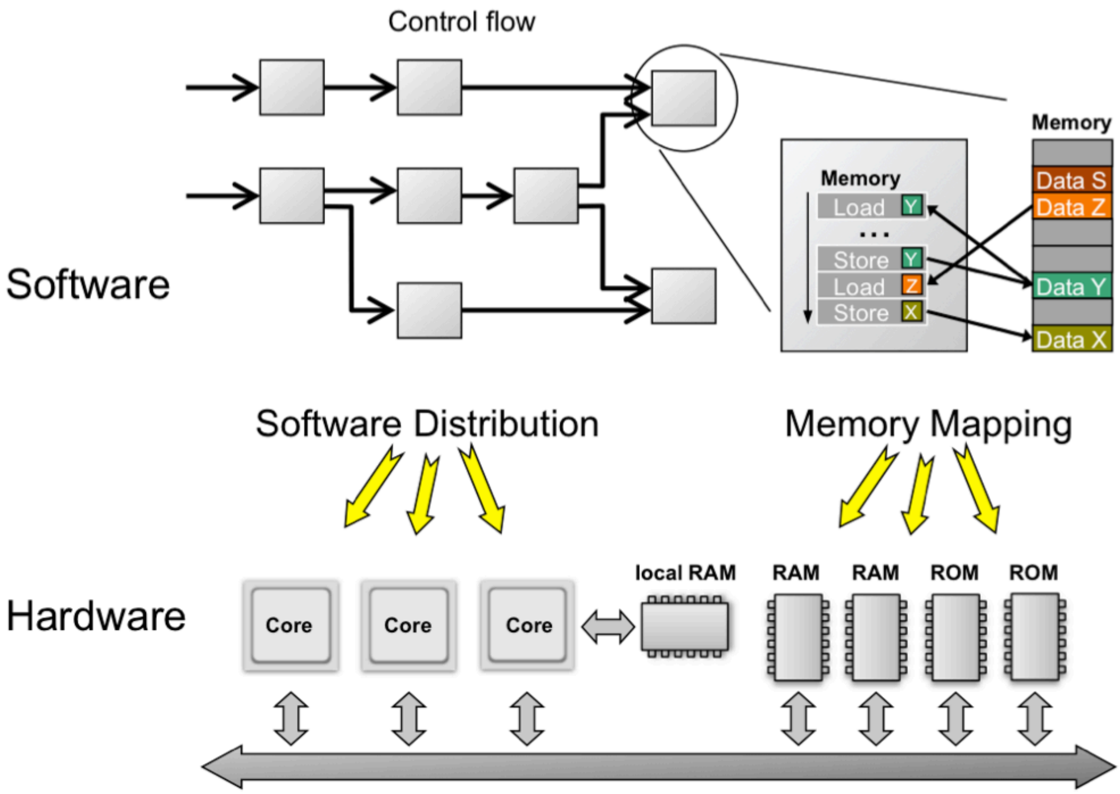

Figure 1: Challanges for Multi-core Systems [Bosch]

Figure 1 shows an abstract view of different challanges in AMALTHEA.

2. Design approach

Compared to a Tool like *Mathwork's Simulink*, the similar open-source approach *Damos* developed by *Itemis AG* [1] as a project partner, shall utilize manufacture's dataflow design processes. Using fragments, synchronous, asynchronous, discrete and continuos communication as well as code generation and simulation this framework supports all necessary functionalities for a consistent development process. Even references to *CReMa*, a requirements engineering tool, can be composed and furthermore bound to a tool-platform like the AUTOSAR architecture basis *Artop* [5].
To effectively distribute software to a multicore-system, certain important steps must be passed in order to gain increasing performance due to essential partitioning, mapping, timing analysis, distribution and consistent optimization through all these processes. According to the Amalthea's initial dataflow- and statechart design in Itemis's Damos and SCT (called Yakindu framework), each block's interface, describing its behavior, possesses a specific calculation. This calculation can be seen as smallest executable units or in terms of AUTOSAR [6] *runnables*. A so called *execution tree* forms an abstract layer between the original model and the partitioning process. It reveals the execution units and dependencies (transitions) between each other. Independent units are shown in separate rows that can be executed in parallel until dependencies cross the corresponding rows. Within the AMALTHEA project, a concept of integrating HW-information to this model, considering dependencies and parallelism shall be developed in order to evaluate execution times as well as to partition runnables into tasks. HW-information in this context includes data about the number of cores, frequency of the cores, shared memory, private memory, type of memory, Operating system, architecture, scheduling, timing constraints, process communication, etc. Additionally specific heuristics evaluating

wether a partition gains increasing performance or to the contrary exposing inadvertent overhead, shall be developed to extend the toolchain to a consistent automatization.
After the optimal partitioning is performed, the *TA Optimizer* [4] shall extend and improve the subsequent mapping process. With the help of a HW-model various scenarios of SW-approaches can be simulated (TA Simulator [4]) respectively the mapping process optimized (*TA Optimizer* [4]). This process not only considers execution times and any HW-information but also SW-architecutre including scheduling algorithms and component structure. Conveying these information back to *Yakindu's Damos* [1], the tool shall handle all gained information in order to perform a HW-specific, timing optimized and highly efficient multicore-system-based code generation. The result shall be code, ready to compile to the HW-platform.
The following picture describes the overall design flow, showing data (models or code), activities and tools each in a separate column.

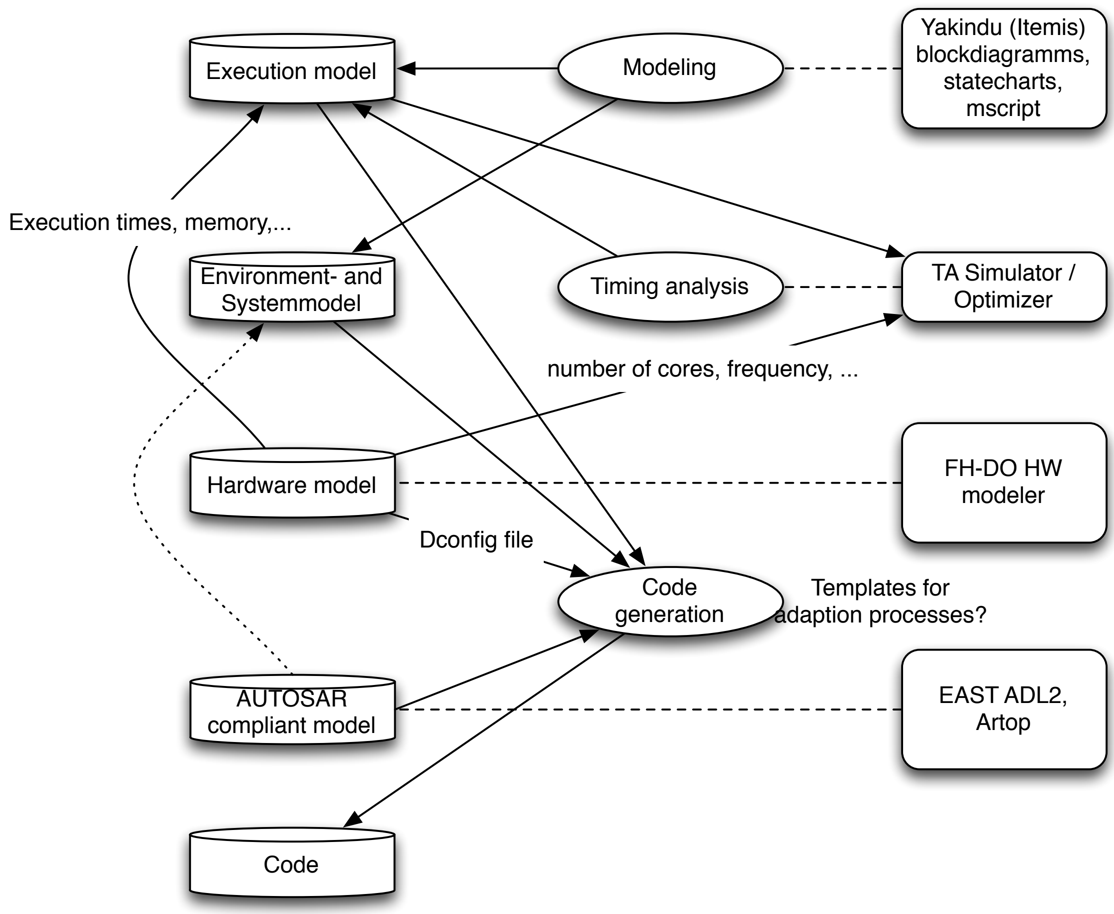

Figure 2: Toolchain designflow

The hardware modeler, a tool created by the Dortmund University of Applied Sciences, shall allow the manual and automatic generation of a hardware model according to specific requirements. In order to automate as much as possible of this process, model importing and profiling will be an inherent part of the modeler as well as interfaces for additional profiling tools. To support further usage of the hardware model, the modeler shall feature an exporter for tool specific extracts with the essential information. These extracts shall provide all necessary information as mentioned above for the code generation as well as the TA tools, essential to perform each processes. EAST-ADL2, Artop, AUTOSAR compliant model and environment- / systemmodels shall provide all necessary information to handle AUTOSAR compliant components / models or further automotive requirements. Finally a common code generation shall combine all models, requirements and system constraints in order to generate all files

needed for a consistent compile process. This process may be seperated into different stages according to logical component's code (ANSI C) and architecture and system code concerning AUTOSAR paradigms. In context of AUTOSAR, and using the corresponding component-model, Damos shall additionally provide the SoftwareComponentDesciption file in order to descibe interfaces and RunTimeEnvironment specific paradigms.

3. Amalthea's Management

Within Amalthea, hierarchical structures are not as much established as in industrial non-research projects. Dividing main project topics into work packages and naming a work package leader for each work package is a necessary step to take control of organizational issues and advantage of efficient coordination. Additionally there is one person representing the Amalthea project in front of the itea2 programm, but no further granularity and all involved people work on similar levels.
For project management and financial issues, there are regular meetings for partner country cooridnators (PCC). Besides these meetings there are cross work package meetings every four months and work package internal meetings to coordinate overlapping issues and prevent redundancy. According to the fact that there are different project partners in each work package, all regular work package meetings are coordinated individually by the work package leaders. In work package three for example, meetings with all work package three partners take place every two months and teleconferences are established for special issues about specific content within the involved people. The people restriction is important in order to keep the conference effective, constructive and the amount of time as short as possible. As soon as people talk on different abstractions respectively on different levels of expertise and detail, especially in a teleconference where people don't see each other, a discussed content may get intransparent and thereby not result in a wished detailed form. To coordinate the level of expertise, workshops are introduced according to topics that influence the development process. Meeting protocols are published into a confluence system (see chapter four) to provide the discussed contents to all partners. Beside the meetings, partners shall publish status updates, describing their work every two weeks for each work package. This task was introduced in order to keep track of ongoing issues and avoid redundant work of different partners. Last but not least, four deliverables spread through the two and a half year project timeframe for each work package form a consistent track of the project progress. Each deliverable contains about a hundred pages and shall sum up work on predefined issues involved in Amlthea's main goals.
All these processes take a lot of time, but are essential for a productive, coordinative and effecitve work on high- and low-level issues in a complex research project.

4. Confluence

Due to the complexity of AMALTHEA's toolchain and the number of persons working on different issues from various places in europe, a specific web-based system is used to exchange data, compose documents, describe work statuses, discuss contents, commonly work on issues, write meeting protocols or post any other relevant information. This system is called confluence [2]. It firstly occurs to be like a wiki system but supports a variety of further functionalities. Confluence provides common ability to review all information, documents and results as well as reduces redundancy, misunderstandings and thereby costs and complexity. Having this system allows access everywhere around the world assuming internet connectivity. The server's data is mirrored which prevents any data loss or failure. Confluence also provides a common independency of commercial office tools, versions or operating systems as it can be accessed via any usual browser. Furthermore document versions and logins are used to always keep track of any document's history providing the ability to restore a earlier version of any data or simply keep track of who is working on what issue. This functionality is extended with the connected

tool JIRA as described in the next chapter. Besides JIRA there is a number of more plugins extending and communicating with confluence. Common and user specific calendars can be used, combined and synchronized with or to any offline calendar. Chatrooms, tables, macros, headings, links, fonts, animations, source-code-highlighting, simplified workflows and many more features can be used in Confluence. Within AMALTHEA, the main aspects concern agreeing on meeting dates, composing commonly predefined deliverables and exchanging experience, innovations or globally relevant information.
The next chapter describes JIRA, another system connected to confluence providing some functionalities not covered by confluence.

5. JIRA

JIRA [3] is an administrative system with various possibilities to organize, track, assign or discuss different issues, problems or tasks as well as generate and analyze time-based diagrams. This especially becomes necessary obtaining a global overview and not to lose track according to different issues, as the number of issues, topics and content may influence and change during the course of a research project. JIRA and Confluence are commercial tools but JIRA is free of charge within a group of less than 21 people. This fact causes JIRA to be only used internally at FH-Dortmund, University of Applied Sciences and Arts.
JIRA focusses on tracking in a way to be always able to reveal timing constraints, possible delays in early development stages and monitor activity. Especially for reporting project statuses, this automated feature provides a quick and easy workflow. Features like the connectivity of a great variety of Issues, code generation, reporting, analysis, featured Add-ons and custom workflows form a flexible easy-to-use environment for state-of-the-art research projects. Connected to confluence the environment supports all necessary features for industrial, commercial or open-Source developments.

6. Conclusion

The research project AMALTHEA takes advantage of different tools and state-of-the-art mechanisms for an effective and collaborative management respectively coordination through a complex field of software development in the automotive research domain. Similar to industrial high level projects, specific people need to identify issues, responsible people and timing constraints to guarantee effective organization. Having these responsible persons for certain well defined issues as well as meetings, workshops and teleconferences, significant benefits without any complications during all project phases and across all work packages can be gained. Working on such complex architectures like AUTOSAR makes these processes indispensable.

7. References

[1] http://www.itemis.de/, itemis AG - a leading IT-supplier for introducing and establishing model based methods, processes and tools for automated software development as well as for realizing company-critical applications, (2012)

[2] http://www.atlassian.com/software/confluence/overview, Confluence - a web based administration system, (2012)

[3] http://www.atlassian.com/software/jira/overview, JIRA – a tracking system for confluence for software development, (2012)

[4] http://www.timing-architects.com/, comprehensive solutions for the evaluation and optimization of the timing, performance, and reliability properties of embedded multicore and manycore systems based on a novel simulation tool, (2012)

[5] http://www.artop.org/, an implementation of common base functionality for AUTOSAR development tools, (2012)

[6] http://www.autosar.org/, a worldwide development partnership of car manufacturers, suppliers and other companies from the electronics, semiconductor and software industry, (2012)

[7] http://www.itea2.org/project/index/view/?project=10015, Model Based Open Source Development Environment for Automotive Multi Core Systems, (2012)